

ATTACHMENT #1

PPP/SLIP DNS DHCP/WINS SNMP Routing

**Master the most
popular inter-
networking protocol
in use today!**



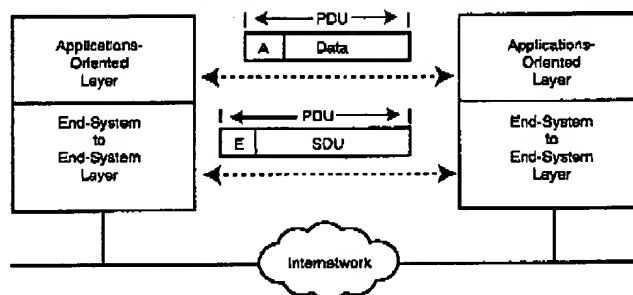
Applet programming Open Systems
OSI Protocols Application integration
FTP Telnet SMTP Routing proto-
cols Configuration issues UNIX and
Linux Windows family Macintosh
NetWare OS/2 Warp PPP/SLIP for
DOS and Windows DNS, NFS, and NIS
SNMP Debugging and configuration
Internetwork management DITCP/WINS

SAMS
PUBLISHING

UNLEASHED

FIGURE 1.11.

Layers at the same level engage in peer talk, as headers provide the vehicle for the exchange of control information at each level. Headers help protocol peers determine the progress of the communications process.



A : Applications-Oriented Layer Header
E : End-System to End-System Layer Header

The top-layer header and user data appear to, and consequently are treated together by, the bottom layer as a *service data unit* (SDU). To the lower layer, an SDU is simply the data that should be provided with the layer's services, with no concern about the details and significance of the data format the SDU might be hiding. This is analogous to your submitting correspondence to the postal service to handle. The postal authority does not concern itself with the actual contents of the envelope being submitted and the details of format that the letter follows.

When a protocol adds its own header information to an SDU, it forms a protocol data unit (PDU). Figure 1.11 illustrates the relationship between an SDU and a PDU. As shown in the diagram, the PDU at the applications-oriented layer becomes the SDU at the end-system-to-end-system layer.

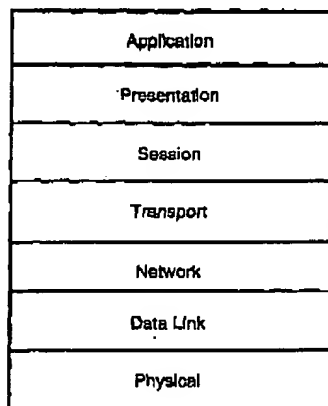
The Open System Interconnection Model

The two-layer model introduced earlier is too simplistic to achieve the recognized benefits outlined in the section titled "The Need for Layered Solutions." This is because each layer combines many functionalities that are independent of each other. For example, the lower end-system-to-end-system layer can be further broken into three layers providing interdependent services. The bottom layer is media specific (that is, mainly physical), addressing concerns such as media type and wiring specifications. The next higher layer is mainly concerned with delivering data link management and flow control services, as well as handling node-to-node communications. (A node could be a computer, bridge, router, or any other network device.) The top layer provides routing services. The same can be said about the applications-oriented layer in the two-layer model. This leads me to introduce the seven-layer model of communications, known as the Open Systems Interconnection (OSI) model.

The OSI model of data communications was developed in 1984 by the International Standardization Organization (ISO). OSI specifies a seven-layer model (see Figure 1.12). In addition to forming the basis of the ongoing development of OSI's own protocols, the model is used by the industry as the frame of reference when describing protocol architectures and

Part I

functional characteristics. This section briefly highlights the general model architecture and the concerns addressed at each layer.

FIGURE 1.12.*The OSI model*

The concept of layering in OSI is governed by two notions: that of service provider and that of the service user. A layer in OSI provides services to the layer above it and uses the services provided by the layer below it. For example, the Network layer in Figure 1.12 provides services to the Transport layer, and it uses services provided by the Data Link layer. A service provider must provide its services while hiding the details on how it is doing it from the service user. In no way should the service user be concerned about how it is getting the service. OSI defines the services that each layer is required to provide to the layer above it. A protocol, or a set of protocols, at any layer is an implementation of those services. Following, you are provided with a brief description of the services handled at each layer.

The Physical Layer

This layer provides the physical transmission service. It accepts data from the Data Link layer in bit streams for the subsequent transmission over the physical medium. At this layer, the mechanical (connector type), electrical (voltage levels), functional (pin assignments), and procedural (handshake) characteristics are defined. RS-232C/D is an example of a Physical layer definition.

The Data Link Layer

This layer is responsible for the reliable transfer of data across the Physical link. Its responsibilities include such functions as data flow control, data frame formatting, error detection, and link management, as discussed earlier in this chapter.

The Network Layer

The Network layer is mainly responsible for providing routing services across the internetwork. It also shields the above layers from details about the underlying network (the network topology and road map) and the routing technology that might have been deployed to connect different networks together.

The Transport Layer

This layer guarantees the orderly and reliable delivery of data between end systems. OSI defines five different protocols at this level, with ranging levels of reliability. The Transport layer also performs additional functions such as data multiplexing and demultiplexing.

The Session Layer

The Session, Presentation, and Application layers are strictly application-oriented layers. They concern themselves with the services useful to applications. No attention is paid at these layers to any of the details governing the data exchange and routing service mechanisms that are well provided at the lower layers.

The Session layer is responsible for establishing, maintaining, and arbitrating the dialogs between communicating applications. It is also responsible for the orderly recovery from failures by implementing appropriate checkpointing mechanisms (see the sections titled "Interprocess Dialog Control" and "Session Recovery" in this chapter for more information).

The Presentation Layer

The Presentation layer is concerned with differences in the data syntax used by communicating applications. This layer is responsible for remedying those differences by resorting to mechanisms that transform the local syntax (specific to the platform in question) to a common one for the purpose of data exchange. ASN.1, the Abstract Syntax Notation, is an example of such common syntax.

The Application Layer

The Application layer provides the engines that drive user applications in an OSI environment. You should make the distinction clear in your mind between the Application layer and end-user applications. To clearly see the distinction, consider the X.400 message-handling system. X.400 defines the engine and protocols that govern message-handling services. As such, X.400 is not the actual messaging application that end users use to deliver mail to the remote users. To do that, users need to install and use mail applications that are X.400-compliant, because only then can the application employ OSI services for the subsequent handling of mail.